Intel® Pentium® 4 Processor with 512KB L2 Cache on .13 Micron Process Signal Integrity Model Usage Guidelines

Revision 1.1

January 2002

Contents

Figures

Tables

Revision History

Rev.	Description	Date
1.1	• Initial revision.	1/7/2002

1. Release Contents

Enclosed are revision 1.1 signal integrity models for the Intel® Pentium® 4 processor with 512KB L2 cache on .13 micron process. These models are based on simulations of the completed I/O buffer design. They are preliminary and subject to change. Refer to the platform design guide for a description of revision states including their corresponding design status and customer expectations.

File	Description
P4p_478_512KB.ibs	Revision 1.1 of the IBIS models
ibischk_log.txt	Log file from the IBIS checker
298641-001.pdf	This document
gowsim_hp	Hewlett-Packard* HP-UX* version of overshoot checker tool
gowsim_ibm	IBM* AIX* version of overshoot checker tool
gowsim_nt.exe	MS-DOS* version of overshoot checker tool
gowsim_sun	Sun* Microsystems Solaris* version of overshoot checker tool
gowsim_linux	Linux* version of overshoot checker tool

2. Introduction

This document describes the models and simulation methods to verify the processor system bus meets signal integrity and timing requirements. The intent is to enable the user to create simulation runs that stress timing and noise margins as prescribed in the *Intel® Pentium® 4 Processor with 512KB L2 Cache on .13 Micron Process* datasheet.

To stimulate signal integrity effects such as ringback or overshoot/undershoot, certain combinations of pull-up and pull-down strengths, edge rate, IO pad capacitance, and package characteristics must be present. The driver and package models to define the worst case corner behaviors are detailed in this document.

Package models are supplied individually and matched with the appropriate driver in simulation to create the necessary corner conditions. The package models represent the high, nominal, and low impedance geometry, and the maximum cross talk condition.

3. Design Space Exploration

To understand the sensitivity and stability of a system under design, Monte Carlo simulation methods are employed. Data analysis focuses on determining the input parameter to output correlation for system variables. The sensitivity of each system variable is assessed and values necessary for creating corner conditions determined.

In the Monte Carlo simulation, all systematic parameters are varied randomly within their tolerance ranges. Variables such as dielectric constant, trace impedance, socket parasitic, and trace line length tolerances are included in the Monte Carlo simulation runs. Initial system characterization runs of approximately 1000 permutations may be sufficient. By running Monte Carlo simulations during the initial design phase, the designer will gain an understanding of the relationships between system variables and system performance.

Table 1 identifies the case simulation corners for system stability to be verified with driver/package/trace model combinations.

Worst Case System Corners	
Corner	Condition
1	Overshoot
2	Undershoot
3	High Side Ring Back
4	Low Side Ring Back
5	Max T-flight
6	Min T-flight

Table 1: Worst Case System Corners

Once system corners and parameter significant lists have been identified, it is recommended that systematic parameter sweep simulations be run with each of the driver/package/trace model significant list combinations. This will allow the designer to define specific simulation corner conditions and parameters for tolerance analysis.

4. Socket Model

The processor socket modeled with discrete inductors, capacitors and resistors is shown in the following figure, Figure 1.

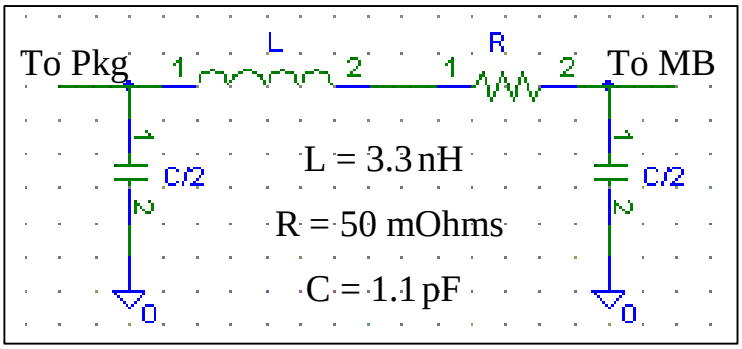

Figure 1: Socket Model

This socket model is currently implemented as a lumped element model. The model is representative of nominal socket electrical parameters. No corner model conditions of high/low impedance were assumed given limited correlation data of possible impedance shift at the time the model was developed.

5. Test Load Correction

System flight times must be corrected to eliminate double counting of the driver rise/fall time, and to correct device timings for system loading effects. It is necessary to subtract the rise/fall delay of the buffer driving into a standard test load from the driver-to-receiver propagation delay in the system.

For each simulation run, the correction factor corresponding to the active driver should be subtracted from the measured flight time. Proper use of the correction factor requires that simulation time zero begins at the moment of voltage change at the driver pad. If this is not the case, the user should define a correction factor based on the standard load provided.

Figure 2 identifies the standard test load for flight time correction. The flight time correction is measured at the pad of the driver driving into the standard test load, and timing measured from voltage change to the voltage reference level, GTLREF.

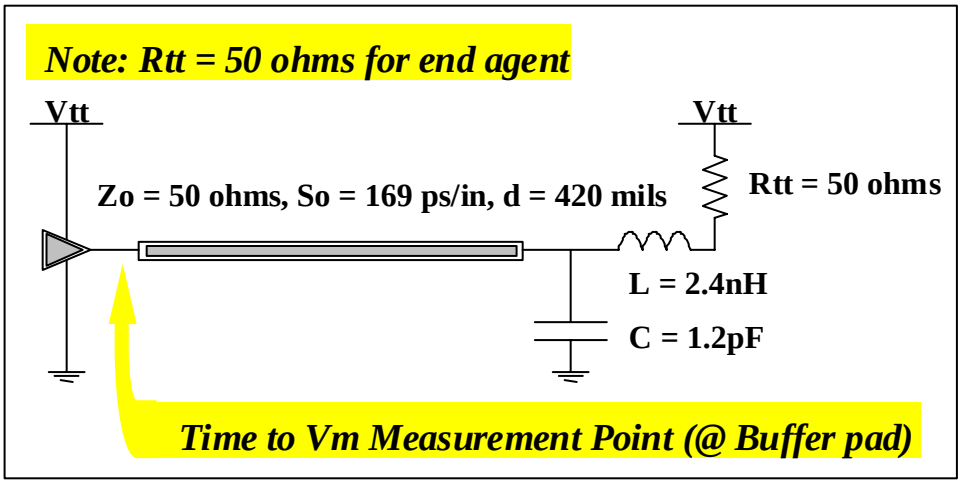

Figure 2: Standard Test Load

Flight time correction and correlation of the driver model to transistor level models can be measured by driving into the standard test load.

6. Simulation and Model Usage

By running Monte Carlo simulations during the initial design phase, an understanding of the relationships between system variables and system performance is used to define the worst case corners. Sensitivity of each system variable was assessed and values necessary for creating corner conditions determined. For each of the case simulation corners, a combination of drivers, package, and system board interconnect parameters was defined among the agents in the system.

7. Appendix

The following is a table of the package model parasitic respective of high/low package model impedance corners.

Model	Lo (H/m)	Co (F/m)	Ro (Ω/m)	Go (mho/m)	Rs (Ω/m)	Gd (mho/m)	Zo(Ω)	So (ns/m)	Note:
MB_Slhigh.rlc	3.924E-07	9.760E-11	3.094E+01	0.000E+00	2.530E-03	6.554E-12	6.341E+01	6.189E-09	SLEM
MB_Sllow.rlc	2.334E-07	1.602E-10	1.683E+01	0.000E+00	2.199E-03	9.550E-12	3.825E+01	6.129E-09	SLEM
CPU_seg1high.rlc	5.860E-07	6.287E-11	5.436E+01	0.000E+00	6.095E-03	2.106E-11	9.654E+01	6.070E-09	SLEM
CPU_seg1low.rlc	1.963E-07	1.922E-10	5.436E+01	0.000E+00	6.095E-03	2.106E-11	3.196E+01	6.141E-09	SLEM
CPU_seg2high.rlc	4.676E-07	8.193E-11	3.448E+01	0.000E+00	4.663E-03	2.254E-11	7.555E+01	6.189E-09	SLEM
CPU_seg1low.rlc	2.067E-07	1.975E-10	3.043E+01	0.000E+00	4.621E-03	2.910E-11	3.235E+01	6.388E-09	SLEM

Table 2: Processor interconnect package model parasitics

Model	Lo (H)	Co (F)	Ro (Ω)	Note:
Die bump	2.000E-11	2.000E-13	2.000E-3	CPU package
PTH via	5.200E-10	3.000E-14	2.100E-3	CPU package
Land pad	1.489E-10	5.000E-15	6.000E-3	CPU package

Table 3: Processor Element Package Model Parasitics

8. Overshoot/Undershoot Checker

Note: This version of the overshoot/undershoot checker utility *gowsim* is only valid for system bus designs using the Intel® Pentium® 4 processor with 512KB L2 cache on .13 micron process. This utility is not valid for use with other processors.

Overview

The Intel® Pentium® 4 processor with 512KB L2 cache on .13 micron process has overshoot/undershoot specifications for system bus signals. These specifications stipulate that a signal at the output of the driver buffer and at the input to the receiver buffer must not exceed a maximum absolute overshoot voltage limit (1.8V) and a minimum absolute undershoot voltage limit (-0.25V) assuming a V_{CC}MAX of 1.50V. Refer to the *Intel® Pentium® 4 Processor with 512KB L2 Cache on .13 Micron Process* datasheet for additional details regarding the undershoot and overshoot. Exceeding these limits will cause damage to the processor. There is also a time dependent, non-linear overshoot requirement above V_{CTERM} and undershoot requirement below GND that is dependent on the amplitude and duration of the overshoot/undershoot.

The overshoot/undershoot checker utility, *gowsim*, has been developed to check for all three of these requirements. This is a post-processing utility that evaluates an OEM generated ASCII data file, which contains time in the first column, followed by space(s), and node voltage in the second column, at the Inputs and Outputs of the processor, checking for any violation of the specifications from simulation perspective. This utility should be run on the results of all simulations to verify there are no violations.

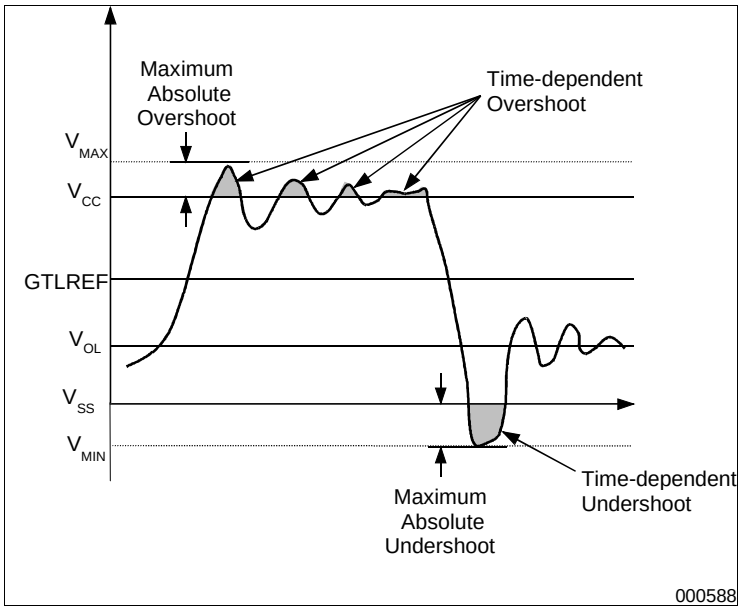

Figure 3: System Bus Signal Waveform Exhibiting Overshoot/Undershoot

Parameter	Description	Specification	Units
V_{CC}MAX	Maximum operating voltage for the processor, equivalent to VccVID	1.50	V
V_{MAX}	Maximum absolute voltage for system bus signals at the input of the receiver buffers	1.80	V
V_{MIN}	Minimum absolute voltage for system bus signals at the input of the receiver buffers	-0.25	V
Overshoot	Time dependent overshoot amount above V_{cc}	[1]	
Undershoot	Time dependent undershoot amount below GND	[1]	

Table 4: Overshoot/Undershoot Parameters

[1] These parameters cannot be specified in absolute terms. They can only be verified by running this *gowsim* tool.

Tool Operation

The program used in the verification process solely relies on the Time-Voltage data file that is extracted from the simulation results.

The ASCII data file format requirements are:

1- The two columns should be time, followed by "space(s)", and followed by voltage.
2- No column headers or empty rows.
3- Voltage should be in Volts; time units are not important so long as they are consistent (this may sound surprising, but think of it as the time units canceling out).
4- The format of the numbers is not important (0.0001 versus 1.000E-4).

The overshoot checker processes the Time-Voltage files and returns whether or not the waveforms meet the overshoot/undershoot requirements (see Table 2 for pass/fail messages). It is possible that a waveform can report all three failure signatures. The verification process typically takes under a minute for all the files extracted from the simulations. ***Gowsim*** is available in separate executables for three UNIX operating systems as well as a MS-DOS executable for Windows* NT workstations:

- IBM* AIX* (compiled under version 4.1)

- Hewlett-Packard* HP-UX* (compiled under version V.10.20)
- Sun* Microsystems Solaris* (compiled under version System 5, Release 4.0)
- Linux*

To execute *gowsim* in a UNIX environment, simply type "./gowsim_*platform*" (where *platform* is either hp, ibm, or sun) and the input file name, as shown below. For Windows* NT, open a DOS window and type "gowsim_nt" and the input file name. The program takes as an argument the name(s) of the *.wvs* file(s):

- `Example: ./gowsim_ibm filename.wvs`
- `Example: gowsim_nt filename.wvs`

The program will output one of the five messages listed in Table 5 below.

Message	Description
PASSED	Signal waveform meets overshoot/undershoot requirements
FAILED: Exceeded maximum voltage of 1.80 V	Signal waveform exceeds the absolute maximum voltage requirement
FAILED: (Exceeded minimum voltage of -0.25V)	Signal waveform exceeds the absolute minimum voltage requirement
FAILED: Overshoot/ undershoot exceeded	Signal waveform exceeds the time dependent overshoot above V_{CTERM}/undershoot below GND
UNKNOWN: (Did not have three complete cycles)	Waveform did not have at least three complete cycles of data transition

Table 5: Gowsim Output Messages and Descriptions

Example Run

Copy the "Time-Voltage" data file(s), for all the nodes, into one directory and run the overshoot checker utility at once:

Input_sim_data.wvs:

```
0.000 0.446
0.010 0.460
0.020 0.479
0.030 0.500
0.040 0.522
0.050 0.546
…
0.660 2.005
0.670 2.010
0.680 2.013
0.690 2.012
0.700 2.009
0.710 2.003
0.720 1.994
…
```

```
> ./gowsim_ibm input_sim_data.wvs
```

The tool prints the evaluation result for each file names on the screen.

Known Issues

The tool has a maximum input of 50,000 lines.

The tool will not accept redundant values in the time column. Take care in using tools that may truncate significant digits such as spreadsheet programs.

Failure Analysis

In the case of a waveform failure, examine the waveform to make sure there is no obvious flaw in the simulation resulting in an excessively high-voltage pulse. If there is no flaw in the data, then the only recourse to meet the process guidelines is to attempt to design the system bus with less overshoot above Vcc and/or less undershoot below GND. Excessive overshoot/undershoot failures can be rectified by decreasing the amplitude of the overshoot/undershoot and reducing the duration of the overshoot/undershoot. Since there is an exponential relation between the signal voltage and the overshoot/undershoot requirement, it is normally far more effective to attempt to reduce the overshoot/undershoot voltage rather than the overshoot/undershoot duration.

Also note that the *gowsim* utility applies pass/fail criteria to the worse case waveform contained in the Time-Voltage input file and, by default, assumes that the magnitude and duration of the worse case waveform occurs on every clock cycle (activity factor = 1). Activity factor can be adjusted by adding "-act <num>" in the command line for *gowsim*.

- Example: gowsim_nt –act 0.01 filename.wvs

9. Disclaimer

The *Intel® Pentium® 4 Processor with 512KB L2 Cache on .13 Micron Process Signal Integrity Models* are solely intended for design validation and do not characterize tester guard-band or full process variation. They are not guaranteed to predict actual or future device behavior. This model suite is not a formal Intel specification and is intended to be a companion to official documentation. In all instances, Intel documented specifications override model predictions.

THESE MODELS ARE PROVIDED "AS IS" WITH NO WARRANTIES OR GUARANTEES WHATSOEVER. Intel EXPRESSLY DISCLAIMS ALL WARRANTIES AND GUARANTEES, INCLUDING ALL WARRANTIES OF MERCHANTABILITY, FITNESS FOR ANY PARTICULAR PURPOSE, NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS, AND ANY WARRANTY OTHERWISE ARISING OUT OF ANY PROPOSAL, SPECIFICATION, OR SAMPLE.

10. Customer Support

For usage or conversion support contact your simulation tool vendor representative.